

有关差分数组

——2021/8/31 Leetcode.1109 航班预订统计

- 题目：

这里有 n 个航班，它们分别从 1 到 n 进行编号。

有一份航班预订表 `bookings`，表中第 i 条预订记录 `bookings[i] = [firsti, lasti, seatsi]` 意味着在从 `firsti` 到 `lasti`（包含 `firsti` 和 `lasti`）的每个航班上预订了 `seatsi` 个座位。

请你返回一个长度为 n 的数组 `answer`，其中 `answer[i]` 是航班 i 上预订的座位总数。

- 示例：

输入: `bookings = [[1,2,10],[2,3,20],[2,5,25]]`, $n = 5$

输出: `[10,55,45,25,25]`

解释：

航班编号	1	2	3	4	5
预订记录 1 :	10	10			
预订记录 2 :		20	20		
预订记录 3 :		25	25	25	25
总座位数:	10	55	45	25	25

因此, `answer = [10,55,45,25,25]`

- 思路：

今天的题是差分数组，我上来直接暴力了，等的就是超时提示😓。我们注意到每一行都是连续的数字，所以能优化的点一定是这里，处理这里可以减少数组的访问次数。但我确实想不出来，后来看答案——差分。老实说我比较头痛这个差分，因为我根本不会下意识往差分那边想🤔。言归正传，差分数组本质还是数组，是个辅助数组，它和前缀和很像，但不同的是，它是有重叠部分的。前缀和就一串数字，挨个加起来当和；差分应用的题，也是连续加（加的形式可能很多种，甚至看不出加，但连续两字至关重要），但它是那一块加，另一块加，等等等等，其中块间有重叠部分。例如本题示例就是如此。

差分数组其实不好用语言来解释它的奥妙，我们先清楚它和前缀和一样，适用于这两种情况：①快速处理区间加减操作②询问区间和问题。然后以后遇到类似问题别忘记它，下意识往这个方向动动脑筋就OK了~

对于本题，嗦那么多不如两个图表来的实际：

- ①按题目老实的处理数据：

航班编号	1	2	3	4	5
预订记录 1	10	10			
预订记录 2		20	20		
预订记录 3		25	25	25	25
总座位数	10	55	45	25	25

- ②用差分思想地处理数据：

航班编号	1	2	3	4	5
预订记录 1	10		-10		
预订记录 2		20		-20	
预订记录 3		25			
总座位数	10	10+45	55-10	45-20	25

像这样，在开始的地方加上座位数，在结束的后一位减去座位数，由于有些数据结束位在最后一位，它没有后一位，故只记录开头，如例子中的25。接下来，从2开始，每一位的总数都等于前一位总数加上自己。其实整个过程很好理解，对于每一位，前面要是还在连续，可以直接继承，要是某个连续的数据停止了，则继承不了它了，要减去它（标在结束后一位就是告诉，下一位数开始，这个数该减去了）。

- 代码：

```
int* corpFlightBookings(int** bookings, int bookingsSize, int* bookingsColSize,
int n, int* returnSize) {
    int* nums = malloc(sizeof(int) * n);
    memset(nums, 0, sizeof(int) * n);
    *returnSize = n;
    for (int i = 0; i < bookingsSize; i++) {
        //开始连续的标记
        nums[bookings[i][0] - 1] += bookings[i][2];
        if (bookings[i][1] < n) {
            //这个数字该结束了
            nums[bookings[i][1]] -= bookings[i][2];
        }
    }
    for (int i = 1; i < n; i++) {
        //继承前面的结果
        nums[i] += nums[i - 1];
    }
    return nums;
}
```

- 后记：

既然遇到了差分，就写上来纪念一下，这个算是技巧一类的。